Title

Improving Carrier-Sense Multiple Access Using Cues of Channel Utilization

Author

Garcia-Luna-Aceves, J.J.

Publication Date

2019

Peer reviewed

Improving Carrier-Sense Multiple Access Using Cues of Channel Utilization

J.J. Garcia-Luna-Aceves
Computer Science and Engineering Department, University of California, Santa Cruz, CA 95064
PARC, Palo Alto, CA 94304
Email: jj@soe.ucsc.edu

***Abstract*—A simple variation of Carrier Sense Multiple Access (CSMA), CUE-CSMA (for Channel Utilization Estimation), is introduced in which the transmission-persistence probability is a function of the perceived average length of idle periods, which is used as a cue of channel utilization. Nodes need not know or estimate the number of nodes in the network. A simple analytical model is used to derive the throughput of CUE-CSMA and compare it with non-persistent and 1-persistent CSMA without having to assume saturation mode as several prior studies have done. The model considers the effect of acknowledgments (ACK) and receive-to-transmit turnaround times. The results clearly show that using estimates of average idle periods as simple cues of channel utilization can provide the benefits of 1-persistent CSMA at light loads and match or outperform non-persistent CSMA at higher loads.**

***Keywords*-Channel access; ALOHA; analytical modeling**

I. Introduction

The original design in Carrier-Sense Multiple Access (CSMA) [9] assumed transmission strategies in which a node with a packet to send decides to transmit based on policies that are independent of the amount of channel congestion. The CSMA transmission strategies can be categorized as non-persistent, 1-persistent, and p-persistent.

According to the non-persistent transmission strategy, a node with a packet to send that detects a busy channel backs off immediately. By contrast, in the 1-persistent strategy, a node with a packet to send that detects carrier continues to persist with its transmission until no carrier is detected in the channel, at which time the node transmits. With the p-persistent strategy, a node with a packet to transmit that finds the channel idle transmits with probability p and a node with a packet to transmit that finds the channel busy repeatedly waits for a propagation delay and tries transmitting again with probability p if the channel is idle or back-off otherwise.

As Section II describes, many approaches have been proposed over the years, to improve on the performance and fairness of CSMA, and p-persistent CSMA in particular, by taking into account channel utilization in the setting of transmission-persistence probabilities and by substituting the basic exponential back-off strategy with the setting of persistence probabilities used by nodes with backlogged transmissions. Many of these prior proposals require nodes to know or estimate the number of nodes attempting to access the channel, which we choose to avoid for the sake of simplicity. Other prior proposals allow nodes to be oblivious of the number of nodes sharing the channel, but require the use of target values of either the probability of packet collisions or the length of idle periods in the channel. The limitation with these approaches is that the target values they use are derived making restrictive assumptions, such as having nodes operate in saturation mode (i.e., all nodes have packets to send at all times), or simplifying assumptions in the derivation of optimal parameter values from which target values are derived.

This paper introduces **CUE-CSMA**, which consists of using measurements of the average idle periods in the channel to change the degree of persistence exerted by nodes that detect a busy channel.

Section III describes how CUE-CSMA operates. Each node individually keeps track of the duration of idle periods it perceives over time, and considers that a reduction of the average idle time beyond a given threshold value μ indicates the onset of channel congestion. A node that has detected a busy channel and receives a local packet to send within ρ seconds from the time when the channel was detected to be busy decides to transmit with a probability φ once the channel becomes idle again. The value of φ used by the node is a a function of the measurements of the average lengths of idle periods. The value of φ tends to 1 as the idle periods last longer than the value μ and tends to 0 as idle periods tend to be shorter than μ.

Section IV presents an analytical model for the computation of the throughput of CUE-CSMA, non-persistent CSMA, and one-persistent CSMA taking into account the use of priority ACKs and the impact of receive-to-transmit turnaround delays that may be longer than propagation delays. Prior work on the performance analysis of the adaptation of persistence probabilities to channel congestion has relied on simulations or has assumed that nodes operate in saturation mode.

Section V provides numerical results comparing the performance of CUE-CSMA, non-persistent CSMA, and one-persistent CSMA. The results show that CUE-CSMA matches the performance of one-persistent CSMA at light loads and outperforms or matches the performance of non-persistent CSMA at medium and high loads. Lastly, Section VI states our conclusions and directions for future work.

II. Related Work

Early analysis of persistent strategies for CSMA [12], [15], [16] focused on 1-persistent and p-persistent variants in which the same values of persistence probabilities are used independently of traffic conditions.

A few variations on transmission persistence have been reported recently (e.g., [6], [7]) that attempt to improve on the performance of traditional persistence strategies with carrier sensing by allowing nodes to persist with their transmissions only for limited periods of time. While such strategies improve on the performance of 1-persistence and p-persistence at light loads, no approach based on constant values of persistence probabilities has been shown to perform better than non-persistent CSMA at high loads.

Several proposals have been made focusing on improving or optimizing the efficiency of CSMA and 802.11 DCF by requiring nodes to estimate [3], [11], [20], [21] or know [1], [5] the number of nodes competing for the channel. Another line of research has focused on improving CSMA by means of cross-layering techniques and physical-layer mechanisms like successive interference cancelation (e.g., [18]).

Many approaches have been advanced to adjust the congestion window (CW) used in the IEEE 802.11 DCF (distributed coordination function) to transmit backlogged packets in order to improve performance (e.g., [2], [4], [8], [10], [13], [19]). A few of these proposals focusing on adaptive CW's in IEEE 802.11 DCF do not require nodes to know or estimate the number of nodes competing for the channel. Of these proposals, we view Idle Sense [8] as the closest in spirit to the overall approach advocated in CUE-CSMA, and hence we discuss tIdle Sense in more detail.

Idle Sense was proposed in the context of the IEEE 802.11 DCF standard, and nodes do not know or estimate the number of nodes competing for the channel. Each node keeps track of the average number of idle slots between two transmission attempts, which we call n_i, and uses that value to compute its CW. The goal is for all nodes to converge to the same common target value for the average number of idle time slots between transmission attempts, which we call n^T, and hence the same CW value. Idle Sense sets $n^T = 5.68$ as a fixed target value that provides satisfactory results when 802.11b DCF parameters are used. The value of n^T is derived from an optimal value of n_i when the number of nodes tends to infinity, , which we denote by n^o. Idle Sense makes n_i converge to n^T using a control algorithm based on the additive increase multiplicative decrease (AIMD) principle applied to the probability with which a node attempts to access the channel, which defines its CW, to ensure fairness among nodes. More specifically, if $n_i > n^T$ the persistence probability is increases by an additive constant, and if $n_i < n^T$ the persistence probability decreases by a multiplicative constant. Accordingly, nodes in Idle Sense increase or decrease their CW's according to the AIMD principle using n^T as a key parameter.

The value of n^o was derived in Idle Sense under the assumption that all transmission periods (successful transmissions, collisions, or idle periods) have the same length, which changes the probabilities of having collisions, a success, or an idle period, given that nodes do persist from one time slot to the next. From the modeling perspective, the work in Idle Sense does not provide an analytical model for the computation of the resulting throughput in the channel. The throughput analysis of Idle Sense [8] was done by simulation experiments.

III. CUE-CSMA

The intent of the proposed Channel-Utilization Estimation (CUE) transmission strategy applied to CSMA is to adapt the rate at which nodes that find the channel busy persist trying to transmit based on channel utiliziation. CUE does this taking advantage of the channel monitoring needed for carrier sensing and virtual carrier sensing.

Different mechanisms can be used to measure channel congestion; however, using the average length of idle periods as the indication of channel congestion is very appealing in the context of carrier sensing. First, nodes using CSMA can determine when the channel is idle or is being used by successful transmissions or transmissions that collide with one another. Second, idle periods tend to be larger at light loads than at high loads, and using long-term averages of their lengths is a simple indicator of congestion. Third, treating a noisy period (during which noise is detected that results from causes other than node transmissions) in the same way as a busy period during which node transmissions take place results in the proper response, which is to discourage transmissions.

We adopt a simpler approach to persistence than the p-persistence scheme first proposed for CSMA by Kleinrock and Tobagi [9]. A node is allowed to persist with its transmission after detecting a busy channel only if the local packet arrives within a *persistence interval* defined to be ρ seconds of the ongoing transmission period.

The local time when the node receives the next local packet to send is denoted by t_p. The local time when a node detects the end of carrier is denoted by t_e. The local time when a node detects carrier following the last end-of carrier time is denoted by t_c. The average of the idle-period lengths estimated by a node is denote by $\tilde{I}$ and is computed using the following simple formula:

$$\tilde{I} = g(|t_c - t_e|) + (1-g)\tilde{I}; \quad 0 < g < 1 \tag{1}$$

The initial value of $\tilde{I}$ is set to 0 and g is a parameter used to assign more or less weight to the length of the last idle period experienced by a node. The *persistence probability* with which a node with a packet to send decides to transmit once the channel becomes idle is denoted by φ. A variety of functions of $\tilde{I}$ can be used to define φ, provided that the result is a probability that changes based on $\tilde{I}$. $\varphi = f(\tilde{I})$ must tend to 0 when $\tilde{I}$ tends to 0 as the channel becomes congested and must tend to 1 as $\tilde{I}$ tends to infinity when the channel is not being used most of the time.

A node that obtains a new local packet to send at time t_p carries out the following steps as part of the channel-access protocol:

1) If the channel is idle at time t_p then:
   a) Update $\tilde{I}$ using Eq. (1) with $t_c = t_p$;
   b) Update φ;
   c) Transmit local packet
2) If the channel is busy at time t_p then:
   a) Compute $TD = |t_p - t_c|$;
   c) If $TD > \rho$
      Then: Apply back-off strategy
      Else:
      Transmit with probability φ at the end of carrier and any required virtual carrier, and back-off if no transmission takes place with probability $1 - \varphi$

The steps stated above are as simple as those that are part of transmission strategies used in prior versions of CSMA and can be used in different types of contention-based channel-access protocols. The key difference with CUE is that the degree with which a node persists with its transmission is a function of the average idle time in the channel, which is learned by the node over time.

We discuss a specific example of φ in Section IV-C as part of our throughput analysis of CUE-CSMA. The design and analysis of persistence probabilities that maximize throughput, and the design and analysis of more sophisticated ways to learn the value of idle periods over time, are the subject of future work.

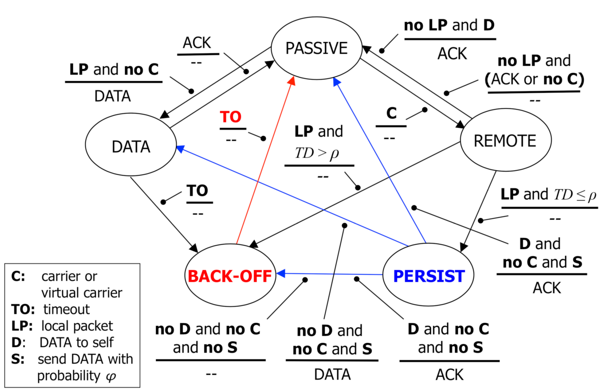

Figure 1. Operation of CUE-CSMA with priority ACKs

Fig. 1 illustrates the operation of CUE-CSMA with priority ACKs using a state machine. CUE-CSMA operates like non-persistent CSMA when a node has to transmit a packet and the channel is idle. On the other hand, if a node receives a local packet to send after it has detected a busy channel, the node transitions to the PERSIST state and carries out the steps described in the previous section as part of the CUE-CSMA protocol based on the values of $\tilde{I}$, φ, and ρ.

A node that receives a local packet to send and detects no carrier transmits its data packet and transitions to the DATA state to wait for an ACK from the receiver. It transitions to the PASSIVE state if an ACK is received or to the BACK-OFF state if the ACK is not received within a timeout period in order to schedule a retransmission.

A node in the PASSIVE state that detects carrier transitions to the REMOTE state and remembers the value of t_c (the local time when carrier was detected). The node transitions back to the PASSIVE state if it has no local packet (LP) to send after either an ACK for another node is received or the end of carrier is detected. The node also transitions to the PASSIVE state after sending its ACK if it receives a data packet for itself while in the REMOTE state.

A node in the REMOTE state that receives a local packet to transmit transitions to the BACK-OFF state if its packet arrives after the persistence interval of ρ seconds (i.e., $TD = |t_l - t_c| > \rho$), and transitions to the PERSIST state otherwise.

A node in the PERSIST state transitions to the BACK-OFF state after detecting the end of carrier if the node decides not to transmit, which occurs with probability $1-\varphi$. In addition, the node transmits its ACK before transitioning to the BACK-OFF state if it receives a data packet for itself from another node while in the PERSIST state.

A node in the PERSIST state that does not decode a data packet for itself from another node transmits its local data packet with probability φ after detecting the end of carrier or virtual carrier and transitions to the DATA state. If the node decodes a data packet for itself while in the PERSIST state, it sends the corresponding ACK and transitions with probability φ to the PASSIVE state, from which it acts on its local data packet.

A node in the BACK-OFF state carries out the steps defined by the back-off strategy. The traditional strategy consists of a node computing a random timeout and transitioning to the PASSIVE state after that time. However, it is important to note that the back-off strategy in CUE-CSMA can be based solely on the setting of persistence probabilities, without forcing any random back-off time at all. Like the basic CSMA design, CUE-CSMA does not inherently require a random back-off time or a congestion window (CW) to be used for retransmissions. This is possible because all nodes accessing the channel have similar estimates of the channel utilization and hence compute similar values for their persistence probabilities, which can be used for both new or retransmitted packets.

It is important to observe that making $\varphi = 0$ or making $\rho = 0$ results in the traditional non-persistent CSMA, because a node detecting carrier always backs off. By the same token, making $\varphi = 1$ and setting ρ to equal the entire period during which the channel is busy results in the traditional one-persistent CSMA strategy.

The design of CUE-CSMA is closely related to the way in which the Idle Sense method operates [8]. However, there are important differences between CUE-CSMA and Idle Sense. CUE-CSMA lets nodes be 1-persistent when average idle periods indicate that less than one packet per packet time is being offered to the channel on average, and makes all nodes reduce their persistence probabilities according to the perceived channel utilization. Idle Sense (as well as other prior schemes aimed at adapting how aggressively nodes attempt to access the channel based on channel conditions) focused on changing the lengths of congestion windows (CW) dynamically. By contrast, there is no inherent notion of a CW in the operation of CUE-CSMA and different approaches are possible based on persistence probabilities that are functions of channel utilization. One approach consists of simply allowing nodes transmit new and retransmitted packets with persistence probabilities that decrease when congestion increases, without incurring back-off delays.

IV. Throughput Analysis

A. Model and Assumptions

We analyze the throughput of CUE-CSMA by extending the approach first described by Sohraby et al. [12] taking into account the effect of receive-to-transmit turnaround times and the use of priority ACKs. According to the model, there is a large number of stations that constitute a Poisson source sending data packets to the the channel with an aggregate mean generation rate of λ packets per unit time. Priority acknowledgments (ACK) are assumed, because they are needed in practice to account for transmission errors not due to multiple-access interference. Each node is assumed to have at most one data packet to sent at any time, which results from the link layer having to submit one packet for transmission before accepting the next packet.

The hardware is assumed to require a fixed turn-around time of ω seconds to transition from receive to transmit or transmit to receive mode for any given transmission to the channel. According to the parameters assumed in IEEE 802.11 DCF, this value may be larger than the propagation delay τ. The transmission time of a data packet is δ and the transmission time for an ACK is α. The channel is assumed to introduce no errors, and multiple access interference (MAI) is the only source of errors. Nodes are assumed to detect carrier and collisions perfectly. We assume that two or more transmissions that overlap in time in the channel must all be retransmitted (i.e., there is no power capture by any transmission), and that any packet propagates to all nodes in τ seconds. Equally important, a channel-access protocol is assumed to operate in steady state, with no possibility of collapse.

To simplify our analysis assuming an infinite population, we assume that a node retransmits a packet after a random retransmission delay. This delay is assumed to be such that all transmissions of data packets can be assumed to be independent of one another. Analyzing the performance of CUE-CSMA when the back-off strategy relies solely on the setting of persistence probabilities requires a model with a finite population (e.g., a Markov chain with states reflecting the number of nodes with packets to send) and is the subject of future work. Our simple model enables us to compare the throughput attained in CUE-CSMA with the throughput in non-persistent and one-persistent CSMA.

B. CUE-CSMA with Priority ACKs

According to the operation of CUE-CSMA, the type of the next transmission period depends on the arrivals that take place during the persistent time ρ of the current transmission period. The utilization of the channel can then be viewed as consisting of *transmission periods* that can be classified based on the number of transmissions at the beginning of a transmission period. This leads to the three-state Markov chain introduced by Sohraby et al. [12]. We call a transmission period of type 0 (TP_0) to be one in which no transmissions take place at the beginning of the transmission period, i.e., an idle period. A transmission period starting with a single transmission is called a transmission period of type 1 (TP_1), and a transmission period that starts with two or more transmissions is called a transmission period of type 2 (TP_2).

Our assumption of steady-state operation results in a homogeneous Markov chain, and the channel must return to any given state within a finite amount of time. We denote by π_i ($i = 0, 1, 2$) the stationary probability of being in state i, i.e., that the system is in a type-i transmission period. The transition probability from state i to state j is denoted by P_{ij}. The average time spent in state i is denoted by T_i. The throughput of the network is then the percentage of time in an average cycle that the channel is used to transmit data successfully, which is

$$S = \frac{\pi_1 \overline{U}}{\pi_0 T_0 + \pi_1 T_1 + \pi_2 T_2} \tag{2}$$

The channel must be in one state at every instant and the channel must transition from one state to another state including itself with probability 1. We observe that $P_{02} = 0$ because arrivals are Poisson distributed and hence there can be no more than one arrival at any instant. On the other hand, because the system is in equilibrium, a packet arrival must occur within a finite time after the channel becomes idle; therefore, $P_{01} = 1$ and $P_{00} = 0$.

The transition from the current transmission period of type 1 or 2 to the next transmission period is solely a function of the number of arrivals during the persistence interval lasting ρ seconds and the value of the current value of the persistence probability φ. This is the case independently of whether one or more transmissions occur at the beginning of the current transmission period, or the success of the current transmission period. Therefore, the type of the next transition period that occurs in CUE-CSMA is independent of whether the current transmission period is of type 1 or type 2, and hence

$$P_{1j} = P_{2j} \qquad \text{for} \qquad j = 0,\ 1,\ 2 \tag{3}$$

We can use the facts stated above and the balance equations for two states of our three-state Markov chain to express the state probabilities as functions of the transition probabilities as follows:

$$\pi_0 + \pi_1 + \pi_2 = 1; \quad P_{j0} + P_{j1} + P_{j2} = 1 \ \text{ for } j = 0, 1, 2;$$
$$\pi_0 = \pi_0(P_{01} + P_{02}) = \pi_1 P_{10} + \pi_2 P_{20} = (\pi_1 + \pi_2)P_{10};$$
$$\pi_2(P_{20} + P_{21}) = \pi_0 P_{02} + \pi_1 P_{12} = \pi_1 P_{12}; \tag{4}$$

The state probabilities can be obtained by solving the system of equations in Eq. (4), which results in:

$$\pi_0 = \frac{P_{10}}{1 + P_{10}}; \ \pi_1 = \frac{P_{10} + P_{11}}{1 + P_{10}}; \ \pi_2 = \frac{1 - P_{10} - P_{11}}{1 + P_{10}} \tag{5}$$

Making use of the fact that $P_{10} + P_{11} = 1 - P_{12}$ in the previous three equations we obtain

$$\pi_0 = \frac{P_{10}}{1 + P_{10}}; \ \pi_1 = \frac{1 - P_{12}}{1 + P_{10}}; \ \pi_2 = \frac{P_{12}}{1 + P_{10}} \tag{6}$$

Substituting Eqs. (6) in Eq. (2) we obtain the following expression of S as a function of transition probabilities P_{10} and P_{12}, $\overline{U}$, and the average times of each transmission period:

$$S = \frac{(1 - P_{12})\overline{U}}{P_{10}T_0 + (1 - P_{12})T_1 + P_{12}T_2} \tag{7}$$

Figure 2 illustrates the type of transmission periods (not to scale) that may occur in CUE-CSMA. The figure shows a sequence of transmission periods with their types indicated by the numbers 0, 1, and 2. As can be observed in the figure, only a TP_1 can be successful and nodes are allowed to persist for ρ seconds after the start of either a TP_1 or a TP_2. Furthermore, only a subset of the nodes that receive packets to send during the persistence interval of ρ seconds transmit at the end of the current transmission period, and that subset is determined by the current value of φ maintained at each node.

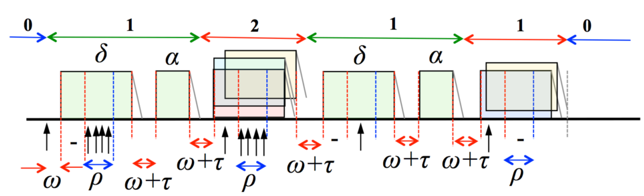

Figure 2. Transmission periods in CUE-CSMA with priority ACKs

The following theorem states the throughput of CUE-CSMA as a function of the persistence probability φ, the length of a persistence interval ρ, and the traffic load.

Theorem 1: The throughput of CUE-CSMA with priority ACKs is

$$S = \frac{(1 + \varphi\lambda\rho)\delta}{(1 + \varphi\lambda\rho)(\omega + \alpha + \tau) + \frac{1}{\lambda}e^{\varphi\lambda\rho} + e^{\lambda(\tau+\omega)}\left(\frac{1}{\lambda} + He^{\varphi\lambda\rho}\right)} \tag{8}$$

where $H = \delta + 2(\omega + \tau) - \frac{1}{\lambda}$

Proof: New arrivals can occur in the first $\tau + \omega$ seconds of a TP_1 or TP_2 because it takes τ seconds for the start of the first transmissions to propagate to all nodes, and a given node that perceives the channel being idle incurs ω seconds transitioning from receive to transmit mode and is deaf during that time. Because arrivals are Poisson with parameter λ, we have that

$$P\{\text{no arrivals in } \tau + \omega\} = e^{-\lambda(\tau+\omega)} \tag{9}$$

The value of $\overline{U}$ is simply the average time in a TP_1 dedicated to successful data. Given that a successful TP_1 occurs with probability $e^{-\lambda(\tau+\omega)}$, we have

$$\overline{U} = \delta e^{-\lambda(\tau+\omega)} \tag{10}$$

The average value of an idle period ($\overline{I} = T_0$) is simply the average inter-arrival time of packets, and given that arrivals are Poisson distributed with parameter λ we have

$$T_0 = 1/\lambda \tag{11}$$

The transition probabilities P_{10} and P_{12} are dependent on the number of nodes that receive packets to send during the ρ seconds of persistence and how many of those nodes decide to persist.

A transition from a TP_1 to a TP_0 requires that either no arrivals occur during the persistence interval of the current transmission period, or that some arrivals did occur in the persistence interval but none of those arrivals persist.

We denote by $(K = 0)$ the event that no nodes with packets to send persist at the end of the current transmission period, and by $(N = n)$ the event that n nodes receive packets to send during the persistence interval of ρ seconds of the current transmission period. Clearly, no node can persist if no packet arrivals occur during the persistence interval of the current transmission period. Therefore,

$$P\{(N = 0)\} = P\{(N = 0) \cap (K = 0)\}$$

For any nonnegative value of n, we also have that

$$P\{(N = n) \cap (K = 0)\} = P\{(K = 0) \mid (N = n)\} \times P\{(N = n)\} \tag{12}$$

Accordingly, the transition probability P_{10} can be expressed as the sum of the probabilities of mutually exclusive events as follows

$$P_{10} = \sum_{n=0}^{\infty} P\{(K = 0) \mid (N = n)\}P\{(N = n)\} \tag{13}$$

Because each node with a packet to send decides to persist with probability φ independently of any other node, we have for all $k \leq n$ that

$$P\{(K=k) \mid (N=n)\} = \binom{n}{k}\varphi^k(1-\varphi)^{n-k} \tag{14}$$

Using the fact that arrivals in the persistence interval are Poisson with parameter λ and substituting Eq. (14) with $k=0$ in Eq. (13) we obtain

$$\begin{aligned} P_{10} &= \sum_{n=0}^{\infty}(1-\varphi)^n \frac{(\lambda\rho)^n}{n!}e^{-\lambda\rho} = e^{-\lambda\rho}\sum_{n=0}^{\infty}\frac{(\lambda\rho(1-\varphi))^n}{n!} \\ &= e^{-\lambda\rho}e^{\lambda\rho(1-\varphi)} = e^{-\varphi\lambda\rho} \end{aligned} \tag{15}$$

A transition from a TP_1 to a TP_2 requires that two or more arrivals occur during the persistence interval of the current transmission period, and that at least two of those arrivals persist. Using a similar approach to the one we used above for P_{10}, the transition probability P_{12} can be expressed as follows

$$P_{12} = \sum_{n=2}^{\infty} P\{(K \geq 2) \mid (N=n)\}P\{(N=n)\} \tag{16}$$

The conditional probability $P\{(K \geq 2) \mid (N=n)\}$ can be expressed in terms of the probabilities for events $(K=0)$ and $(K=1)$ conditioned on event $(N=n)$ as follows

$$\begin{aligned} P\{(K \geq 2) \mid (N=n)\} =& 1 - P\{(K=0) \mid (N=n)\} \\ & - P\{(K=1) \mid (N=n)\} \end{aligned} \tag{17}$$

Using the fact that each node with a packet to send decides to persist with probability φ independently of any other node and Eq. (14) with $K=0$ and $K=1$, we have

$$P\{K \geq 2 \mid N=n\} = 1-(1-\varphi)^n - n\varphi(1-\varphi)^{n-1} \tag{18}$$

Substituting Eq. (18) in Eq. (16) and given that arrivals in the persistence interval are Poisson with parameter λ we obtain

$$\begin{aligned} P_{12} =& \sum_{n=2}^{\infty}\frac{(\lambda\rho)^n}{n!}e^{-\lambda\rho} - \sum_{n=2}^{\infty}(1-\varphi)^n\frac{(\lambda\rho)^n}{n!}e^{-\lambda\rho} \\ & - \sum_{n=2}^{\infty} n\varphi(1-\varphi)^{n-1}\frac{(\lambda\rho)^n}{n!}e^{-\lambda\rho} \\ =& \left(e^{\lambda\rho} - \left[e^{\lambda\rho}e^{-\varphi\lambda\rho} + \varphi\lambda\rho\right] - \varphi\lambda\rho\left(e^{\lambda\rho}e^{-\varphi\lambda\rho} - 1\right)\right)e^{-\lambda\rho} \end{aligned} \tag{19}$$

Simplifying the previous expression for P_{12} we obtain

$$P_{12} = 1 - e^{-\varphi\lambda\rho} - \varphi\lambda\rho e^{-\varphi\lambda\rho} \tag{20}$$

The results for P_{10} and P_{12} in Eqs. (15) and (20) are rather intuitive given the Poisson-arrival assumption of our model. Each arrival that takes place during the persistence interval of ρ seconds is "selected" with probability φ to persist independently of other arrivals, which amounts to decomposing the Poisson source into two independent streams defined by φ and $1-\varphi$. Eqs. (15) and (20) can be viewed as a consequence of this, because it is well known that decomposing a Poisson process with parameter λ into two or more independent streams results in each stream randomly selected with probability p being a Poisson process with parameter $p\lambda$.

Substituting Eqs. (10), (11), (15), and (20) in Eq. (7) we have

$$S = \frac{(1+\varphi\lambda\rho)\delta e^{-\lambda(\omega+\tau)}}{\frac{1}{\lambda} + (1+\varphi\lambda\rho)T_1 + (e^{\varphi\lambda\rho} - (1+\varphi\lambda\rho))T_2} \tag{21}$$

The lengths of transmission periods of type 1 and 2 are functions of the time between the first and the last transmission in the transmission period, which is a random variable Y that can assume values between 0 and $\tau+\omega$. If the time period between the start of the the first and the last data packets in a collision interval equals y seconds, then there are no more packet arrivals in the remaining time of the vulnerability period of the first packet of the collision interval, i.e., $\omega+\tau-y$ seconds. Accordingly, $P(Y \leq y) = F_Y(y) = e^{-\lambda(\omega+\tau-y)}$. Therefore, given that Y assumes only non-negative values, the average value of Y equals

$$\begin{aligned} \overline{Y} &= \int_0^{\infty}(1-F_Y(t))dt = \int_0^{\omega+\tau}\left(1-e^{-\lambda(\omega+\tau-t)}\right)dt \\ &= \quad \omega+\tau - \frac{1-e^{-\lambda(\omega+\tau)}}{\lambda} \end{aligned} \tag{22}$$

If we consider the turnaround delay incurred by a transceiver that detects an idle channel, it takes ω seconds for a node to start transmitting after detecting an idle channel or the end of carrier in a TP_1 or a TP_2. Hence, a delay of ω seconds can be assumed at the beginning of a TP_1 or a TP_2.

Because a TP_2 starts with two or more transmissions, no success can occur in it, and hence it consists of overlapping packets that cannot be decoded by the intended receivers. Accordingly, the average length of a TP_2 equals $\omega+\overline{Y}+\delta+\tau$, and substituting the value of $\overline{Y}$ we have

$$T_2 = \delta + 2(\omega+\tau) - \frac{1-e^{-\lambda(\omega+\tau)}}{\lambda} \tag{23}$$

A TP_1 succeeds if no arrivals occur during its vulnerability period, which occurs with probability $e^{-\lambda(\tau+\omega)}$. If successful, the transmission period includes an ACK, and otherwise it consists of overlapping data packets as in a TP_2. As we have stated before, there can be no more than one arrival at any instant because arrivals are Poisson distributed. This means that the case of $Y=0$ occurs only when the transmission period succeeds, i.e., when the first and the last transmission in the period are the same. Therefore,

$$T_1 = T_2 + e^{-\lambda(\tau+\omega)}(\omega + \alpha + \tau) \tag{24}$$

Substituting Eq. (23) in Eq. (24) we obtain

$$T_1 = \delta + 2(\omega+\tau) - \frac{1}{\lambda} + e^{-\lambda(\omega+\tau)}\left(\omega + \alpha + \tau + \frac{1}{\lambda}\right) \tag{25}$$

Substituting Eqs. (23) and (25) in Eq. (21) we obtain Eq. (8) after some simplification. ■

C. Using a Simple Persistence-Probability Function

The persistence probability φ should be a function of $\tilde{I}$ that has a value of 1 or close to 1 when traffic load is light and a value that quickly decreases as traffic load increases to the point that packet collisions are likely.

Let μ be a threshold value for the length of an average idle period indicating that the channel is becoming congested when $\tilde{I} < \mu$. The following function for φ is a simple example of how persistence probabilities can be adapted to channel utilization using a minimum length of average idle periods as the indicator of congestion:

$$\varphi = \begin{cases} 1 & \text{if } \tilde{I} \geq \mu \\ (\tilde{I}/\mu)^{\beta} & \text{if } \tilde{I} < \mu \end{cases} \tag{26}$$

Independently of how nodes persist accessing the channel once they detect carriers, packet collisions become very likely when on average there are multiple arrivals per packet time. Accordingly, setting $\mu = \delta$ makes sense when all packets are of the same length. In practice, μ would be equal to the duration of an average packet.

The larger the values of β are, the less aggressive a node becomes once it detects that the average length of idle periods signals the onset of channel congestion. Therefore, we assume in our analysis that $\beta = 2$ to make nodes more aggressive than in non-persistent CSMA around traffic loads when average idle periods last one packet time, but not aggressive enough to reduce throughput at high loads as is the case with one-persistent CSMA.

For comparison purposes, we assume that nodes have a good estimate of the average length of idle periods, which means that $\tilde{I} = \bar{I}$. Because we assume Poisson arrivals with parameter λ and a node with a packet to send transmits with probability 1 after detecting an idle channel, we have that $\tilde{I} = 1/\lambda$. Given these considerations, we have:

$$\varphi(\lambda) = \begin{cases} 1 & \text{if } 1/\lambda \geq \delta \\ (1/\lambda\delta)^{\beta} & \text{if } 1/\lambda < \delta \end{cases} \tag{27}$$

D. Non-Persistent CSMA with Priority ACKs

The throughput of non-persistent CSMA with priority ACKs was obtained by Kleinrock and Tobagi [17] for the case in which turnaround latencies are negligible.

Figure 3 illustrates the transmission periods in non-persistent CSMA with priority ACKs when turnaround latencies increase the vulnerability period of a data packet. It can be observed from the figure that a transmission period can either be idle or start with a single transmission, i.e., there are no transmission periods of type 2.

The following theorem states the throughput of non-persistent CSMA with priority ACKs taking into account the effect of turnaround latencies.

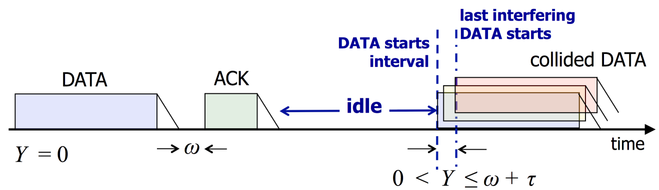

Figure 3. Transmission periods in non-persistent CSMA with priority ACKs

Theorem 2: The throughput of non-persistent CSMA with priority ACKs is

$$S_{NP} = \frac{\delta}{\omega + \alpha + \tau + \frac{1}{\lambda} + e^{\lambda(\omega+\tau)}(\delta + 2\omega + 2\tau)} \tag{28}$$

Proof: The result follows directly from Theorem 1 by setting $\rho = 0$ in Eq. (8). ■

The result of the previous theorem should be obvious given that the state machine of CUE-CSMA renders the non-persistent version of CSMA when $\rho = 0$.

E. One-Persistent CSMA with Priority ACKs

The results for one-persistent CSMA (1P-CSMA) derived by Kleinrock and Tobagi [9] and Sohraby et al. [12] assume that ACKs are sent in a secondary channel in zero time without interference, and that turnaround latencies are negligible.

In contrast to CUE-CSMA in which the persistence interval ρ is a constant, the length of a persistence interval in 1P-CSMA is a random variable. More specifically, the length of ρ in 1P-CSMA is the remaining length of the current transmission period after the first $\tau+\omega$ seconds have elapsed and nodes can detect carrier.

Due to space limitations, we provide an approximated analysis of the performance of 1P-CSMA by assuming that nodes that detect carrier persist with probability 1 during a persistence interval of length δ. The following theorem states the result.

Theorem 3: The throughput of 1-P CSMA with priority ACKs is

$$S_{1P} \leq \frac{(1+\lambda\delta)\delta}{(1+\lambda\delta)(\omega+\alpha+\tau) + \frac{1}{\lambda}e^{\lambda\delta} + e^{\lambda(\tau+\omega)}\left(\frac{1}{\lambda} + He^{\lambda\delta}\right)} \tag{29}$$

where $H = \delta + 2(\omega + \tau) - \frac{1}{\lambda}$

Proof: The right-hand side (RHS) of Eq. (29) follows from Theorem 1 by setting $\varphi = 1$ and $\rho = \delta$ in Eq. (8). This is an an upper bound on the performance of 1P-CSMA because the result assumes that the persistence interval of a transmission period is only δ seconds but in 1P-CSMA nodes persist for $Y + \delta + \tau$ seconds during a failed transmission period and for $\delta + \alpha + \omega + 2\tau$ seconds during a successful transmission period, where $\overline{Y}$ is given in Eq. (22). It follows that, for a given value of λ, the result assumes fewer persisting transmissions from nodes that have detected carrier during a collision interval or a successful transmission interval. Consequently, the RHS of Eq. (29) states higher throughput values than 1P-CSMA can attain as λ increases. ■

V. Numerical Results

The throughput attained by a channel-access protocol is a function of the physical layer and link layer. However, the physical-layer overhead is roughly the same for all the channel-access protocols we consider and for simplicity, other than turnaround times we do not consider the PHY-level overhead in our comparison of channel-access protocols.

We assume a channel data rate of 1 Mbps even though higher data rates are common today; this is done just for simplicity given that the actual data rates of transmitted packets do not impact the relative performance differences among transmission strategies. We assume link-level lengths of signaling packets similar to those used in IEEE 802.11 DCF. However, we assume that an ACK is 40 bytes. We assume that ω is an order of magnitude longer than the propagation delay, which results in lower throughput, because the vulnerability period of a data packet is that much longer.

We normalize the results to the length of a data packet by making $\delta = 1$, $G = \lambda \times \delta$, and $a = \tau/\delta$; and by using the normalized value of each other variable, which equals its ratio with δ. Physical distances are around 500 meters, and the duration of a data packet is 1500 bytes, which is an average-length IP packet and takes 0.012s to transmit at 1 Mbps. We use a normalized propagation delay of $a = 1 \times 10^{-4}$.

We compare the throughput of CUE-CSMA with non-persistent CSMA (NP-CSMA) and one-persistent CSMA (1P-CSMA) when priority ACKs are used in each of them.

Fig. 4 shows the throughput (S) versus the offered load (G) for the three channel-access protocols based on Eq. (8) for CUE-CSMA using Eq. (27) for the values of φ and a function of λ, Eq. (28) for non-persistent CSMA, and Eq. (29) for one-persistent CSMA. For simplicity we assume that $\rho = \delta$. As we have stated, we assume that nodes have an accurate estimate of the average length of idle periods to focus on the impact that the persistence-probability function has on performance compared to fixed values of persistence.

The results for 1P-CSMA with ACKs show that persisting with probability 1 independently of channel congestion for the entire duration of transmission periods becomes too aggressive even at moderate loads, which negates the benefits of having persistence. On the other hand, the results reported in [7] show that limited-persistence strategies perform worse than 1-P CSMA at light loads because nodes are not aggressive enough, and perform worse than NP-CSMA at high loads because nodes are too aggressive.

CUE-CSMA is the first persistence strategy that can perform better than or equal to a non-persistence or one-persistent strategy at any value of offered load. This is the case because CUE-CSMA takes full advantage of persistence at light to moderate offered loads and behaves more and more like NP-CSMA as the offered load increases beyond one packet per packet time on average. Fig. 5 shows the throughput gains derived from adapting transmission persistence in CUE-CSMA compared to 1P-CSMA and NP-CSMA.

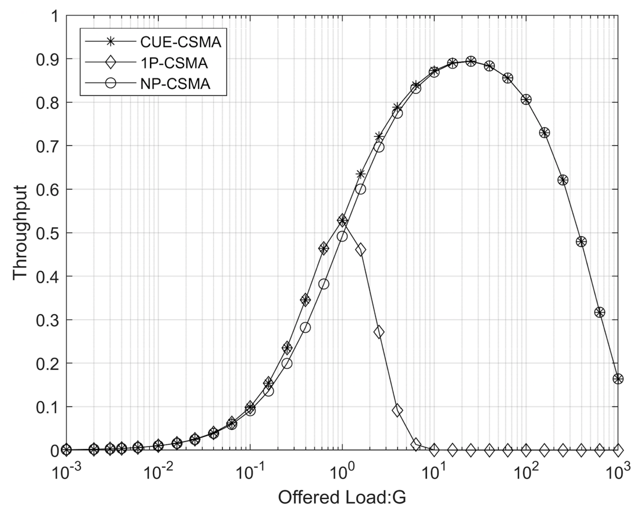

Figure 4. S vs. G for NP-CSMA, 1P-CSMA, and CUE-CSMA

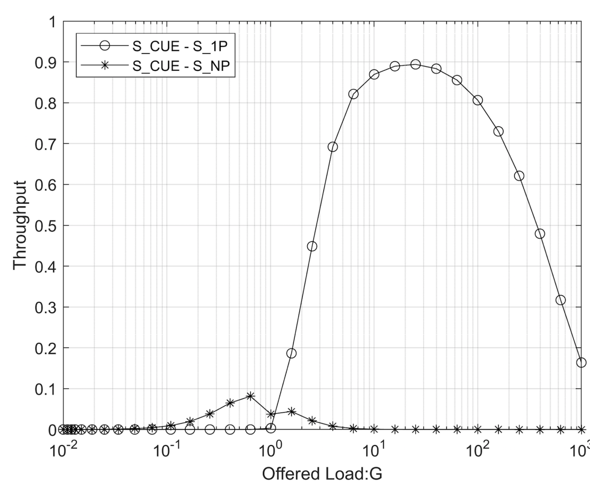

Figure 5. Throughput gain attained with CUE-CSMA over NP-CSMA and 1P-CSMA

VI. CONCLUSIONS AND FUTURE WORK

We introduced CUE-CSMA, a simple approach to adapting transmission persistence in channel-access protocols based on CSMA by using perceived channel utilization indicated that estimates of the length of average idle periods. Nodes learn over time the average length of idle periods and use that knowledge as their cue to adjust the probabilities with which they persist with their transmissions after detecting a busy channel.

We analyzed the performance of CUE-CSMA, non-persistent CSMA and 1-persistent CSMA using a simple analytical model that can be used for the three protocols. The results of our analysis shows that CUE-CSMA provides better throughput than 1-persistent CSMA and non-persistent CSMA for any traffic-load condition.

CUE can be applied to any channel-access protocol that uses carrier sensing or virtual carrier sensing. Our immediate future work in this area focuses on defining how CUE should operate in the context of IEEE 802.11, providing an analytical model for a finite node population to study the effect of substituting the traditional exponential back-off strategy with the simple adaptation of persistence probabilities, and complementing the analysis with simulations.

Other promising areas for future research related to CUE-CSMA include: (a) designing more sophisticated functions for the persistence probability than the simple function we have introduced; (b) using additional information for persistence strategies, such as the state of the channel-access protocol; and (c) designing learning mechanisms for nodes to quickly update their estimates of channel utilization, which are critical for performance and stability of the channel-access protocols.

ACKNOWLEDGMENTS

This work was supported in part by the Jack Baskin Chair of Computer Engineering at UC Santa Cruz. Najmeh Mashhadi provided valuable help in the presentation of numerical results.

REFERENCES

[1] H. Anouar and C. Bonnet, "Optimal Constant-Window Backoff Scheme for IEEE 802.11 DCF in Single-Hop Wireless Networks Under Finite Load Conditions," *Wireless Personal Communications*, Dec. 2007.

[2] J. Barcelo et al., "Dynamic P-Persistent Backoff for Higher Efficiency and Implicit Prioritization," *Proc. JITEL '08*, 2008.

[3] G. Bianchi and I. Tinnirello, "Kalman Filter Estimation of the Number of Competing Terminals in an IEEE 802.11 Network," *Proc. IEEE INFOCOM '03*, 2003.

[4] L. Bononi, M. Conti, and E. Gregori, "Runtime Optimization of IEEE 802.11 Wireless LANs Performance," *IEEE Transactions on Parallel and Distributed Systems*, 2004.

[5] F. Cali, et al., "Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit," *IEEE/ACM Transactions on Networking*, 2000.

[6] J.J. Garcia-Luna-Aceves, S. Thompson, and J. Stern, "Time-Based Persistence in Channel-Access Protocols with Carrier Sensing," *Proc. IEEE IPCCC 2017*, Dec. 2017.

[7] J.J. Garcia-Luna-Aceves, "A State-Aware Persistence Strategy for Multiple Access Protocols with Carrier Sensing," *Proc. IEEE WCNC '18*, 2018.

[8] M. Heusse et al., "Idle Sense: An Optimal Access Method for High Throughput and Fairness in Rate Diverse Wireless LANs," *Proc. ACM SIGCOMM '05*, 2005.

[9] L. Kleinrock and F. A. Tobagi, "Packet Switching in Radio Channels: Part I - Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics," *IEEE Trans. Commun.*, 1975.

[10] Y. Kwon, Y. Fang, and H. Latchman, "A Novel MAC Protocol with Fast Collision Resolution for Wireless LANs," *Proc. IEEE INFOCOM '03*, 2003.

[11] A. Lopez-Toledo et al.,"Adaptive Optimization of IEEE 802.11 DCF Based on Bayesian Estimation of the Number of Competing Terminals," *IEEE Trans. on Mobile Computing*, Sept. 2006.

[12] K. Sohraby et al., "Comments on Throughput Analysis for Persistent CSMA systems, *IEEE Trans. Commun.* January 1987.

[13] N. O. Song et al., "Enhancement of IEEE 802.11 Distributed Coordination Function with Exponential Increase Exponential Decrease Backoff Algorithm," *Proc. IEEE VTC '03*, 2003.

[14] R. Sutton and A.G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 1998.

[15] H. Takagi and L. Kleinrock, "Throughput Analysis for Persistent CSMA Systems," *IEEE Trans. Commun*, July 1985.

[16] F. A. Tobagi and V.B. Hunt, "Performance Analysis of Carrier Sense Multiple-Access with Collision Detection," *Comput. Networks*, Oct./Nov. 1980.

[17] F. Tobagi and L. Kleinrock, "The Effect of Acknowledgment Traffic on the Capacity of Packet-Switched Radio Channels," *IEEE Trans. Commun.*, 1978.

[18] X. Wang et al., "Design and analysis of Cross-Layer Tree Algorithms for Wireless Random Access," *IEEE Transactions on Wireless Communications*, March 2008.

[19] T. You, C.-H. Yeh, and H. Hassanein, "A New Class of Collision-Prevention MAC Protocols for Ad Hoc Wireless Networks," *Proc. IEEE ICC '03*, 2003.

[20] L. Zhao et al., "Using Incompletely Cooperative Game Theory in Mobile Ad Hoc Networks," *Proc. IEEE ICC '07* 2007.

[21] L. Zhao, J. Zhang, and H. Zhang, "Using Incompletely Cooperative Game Theory in Wireless Mesh Networks," *IEEE Network*, 2008.